Compressive Recovery of Sparse Precision Matrices

Etienne Lasalle

with

Titouan Vayer

Rémi Gribonval

Paulo Gonçalves

Journées de Statistiques, Bordeaux.
May 2024.

ENS DE LYON

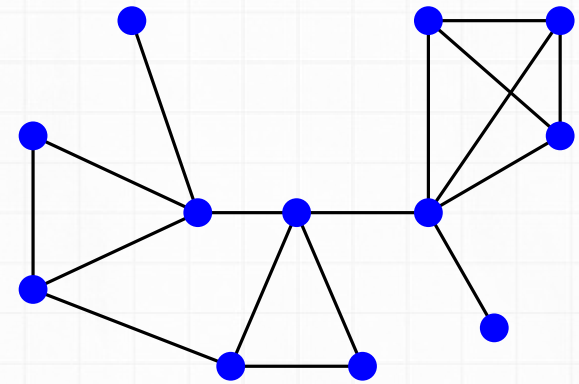
etienne.lasalle@ens-lyon.fr

The problem

Data

Find the interactions

between the d variables



$$\mathbf{x}_1, \ldots, \mathbf{x}_n \overset{i.i.d}{\sim} \mathcal{N}(0, \mathbf{\Sigma} = \mathbf{\Theta}^{-1})$$

Gaussian model

$\mathbf{\Theta}$: precision matrix

(Conditional dependencies)

Framework:

- high dimension : d is large
- Sparsity in $\mathbf{\Theta}$

GLASSO approach [FHT08], [BGA08]

Naive estimator of $\boldsymbol{\Theta}$ (MLE): $\widehat{\boldsymbol{\Sigma}}^{-1}$ (Not sparse)

Graphical Lasso

i.e., ℓ_1-penalized maximum likelihood estimator

$$\widehat{\boldsymbol{\Theta}}_{\mathrm{GL}} \triangleq \underset{\boldsymbol{\Theta} \succ 0}{\arg\min} \left\{ -\log\det(\boldsymbol{\Theta}) + \langle \widehat{\boldsymbol{\Sigma}}, \boldsymbol{\Theta} \rangle + \lambda \|\boldsymbol{\Theta}\|_{1,\mathrm{off}} \right\},$$

$$\widehat{\boldsymbol{\Sigma}} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^T$$

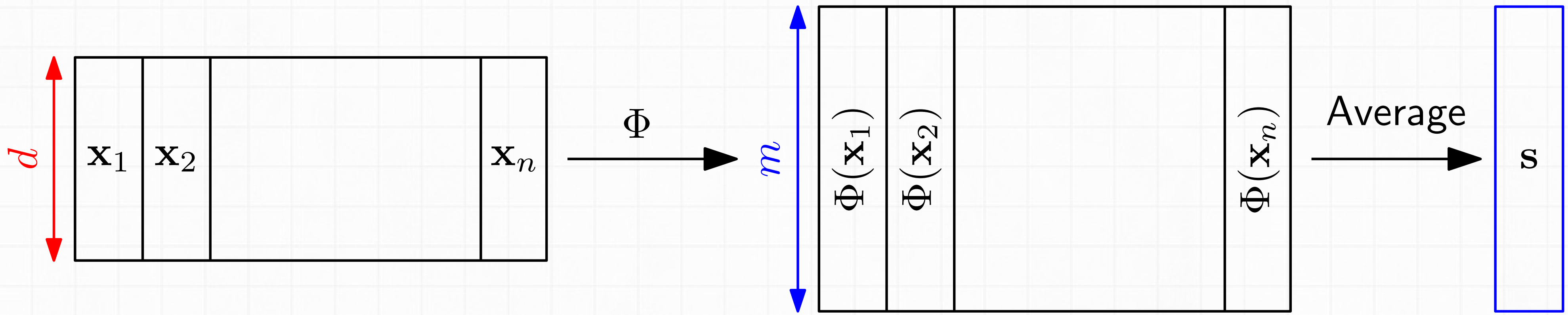
$\|\boldsymbol{\Theta}\|_{1,\mathrm{off}} = \sum_{i<j} |\boldsymbol{\Theta}_{ij}|$

λ : regularization parameter

Memory cost : $\mathcal{O}(d^2)$ (storage of $\widehat{\boldsymbol{\Sigma}}$).
Computational cost : $\mathcal{O}(d^3)$.

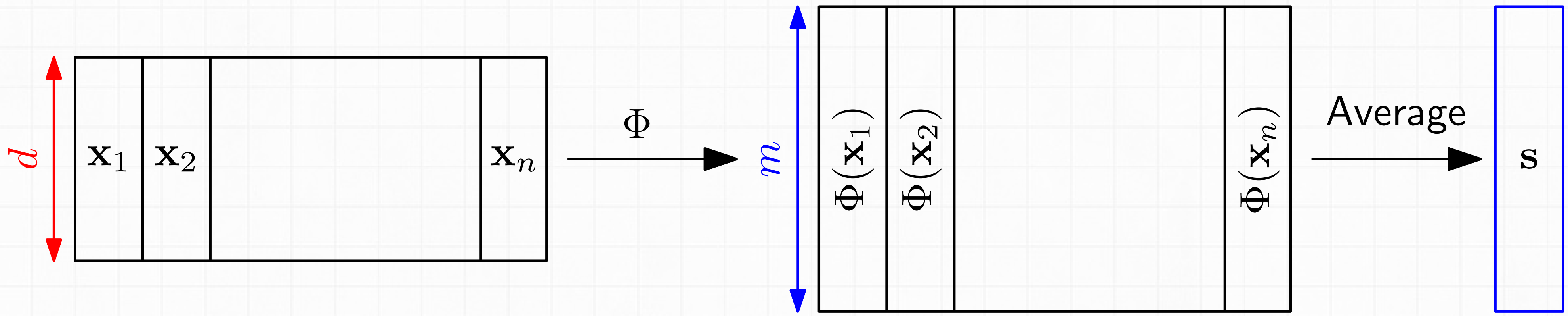
[FHT08] Friedman *et. al.*, *Sparse inverse covariance estimation with the graphical lasso*, (2008)
[BGA08] Banerjee *et al.*, *Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data*, (2008)

Sketching approach (compression) [GCK+21]



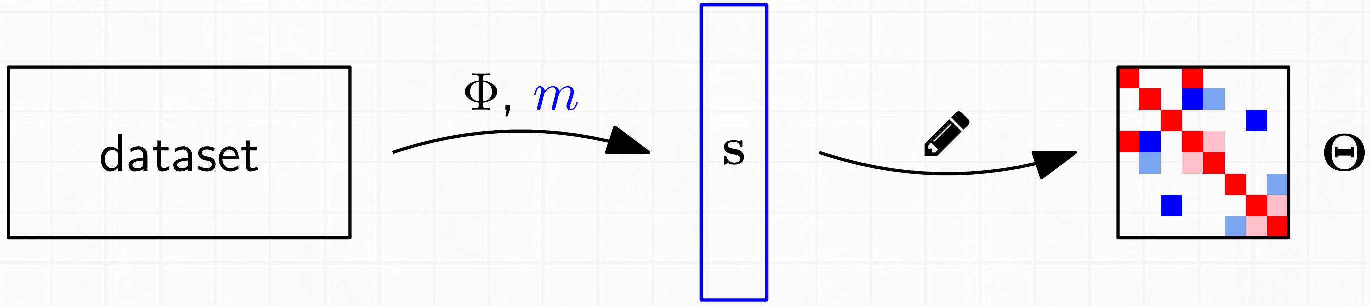
[GCK+21] Gribonval *et al.*, *Sketching Data Sets for Large-Scale Learning: Keeping only what you need*, (2021)

Sketching approach (compression) [GCK+21]



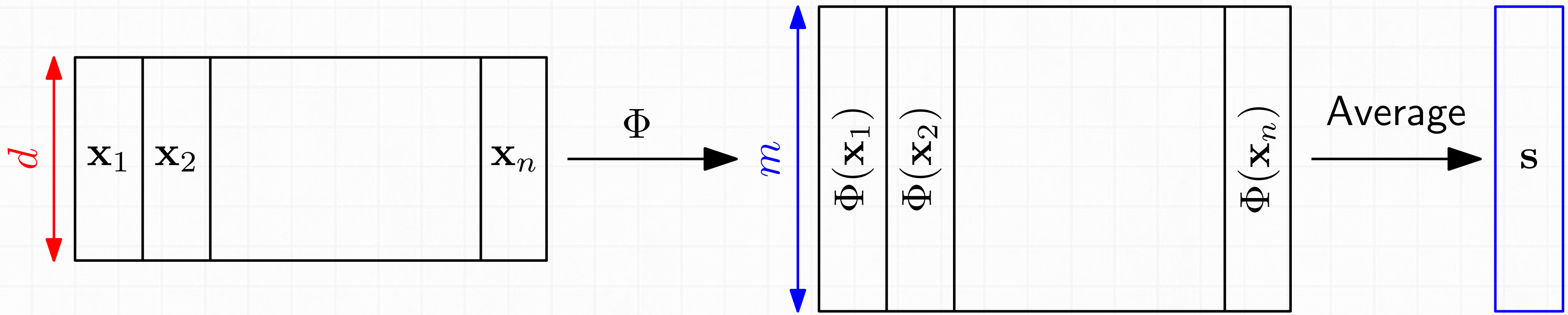
Questions :

- How to define $\mathbf{s}$? (Ideally : $\|\boldsymbol{\Theta}\|_0 \lesssim m \ll d^2$)
- Sketch : is it efficient (computation and memory)?
- Practical decoder?
- Is it efficient?



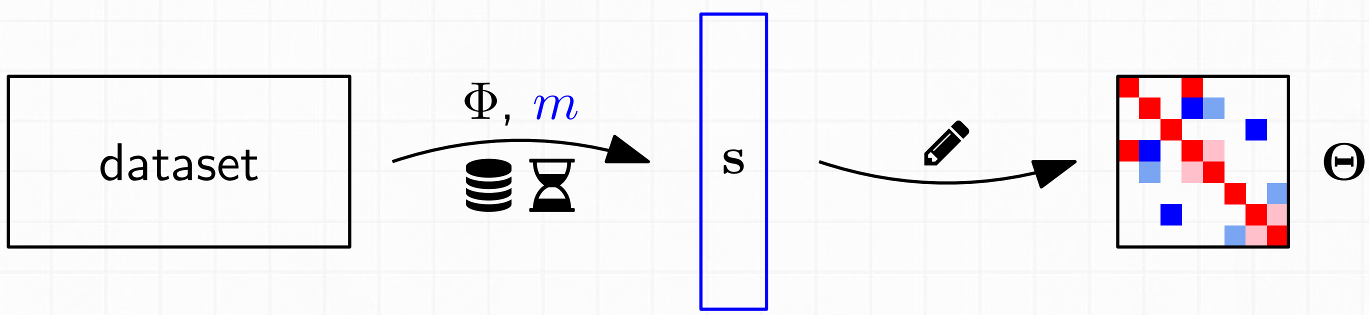
[GCK+21] Gribonval *et al.*, *Sketching Data Sets for Large-Scale Learning: Keeping only what you need*, (2021)

Sketching approach (compression) [GCK+21]



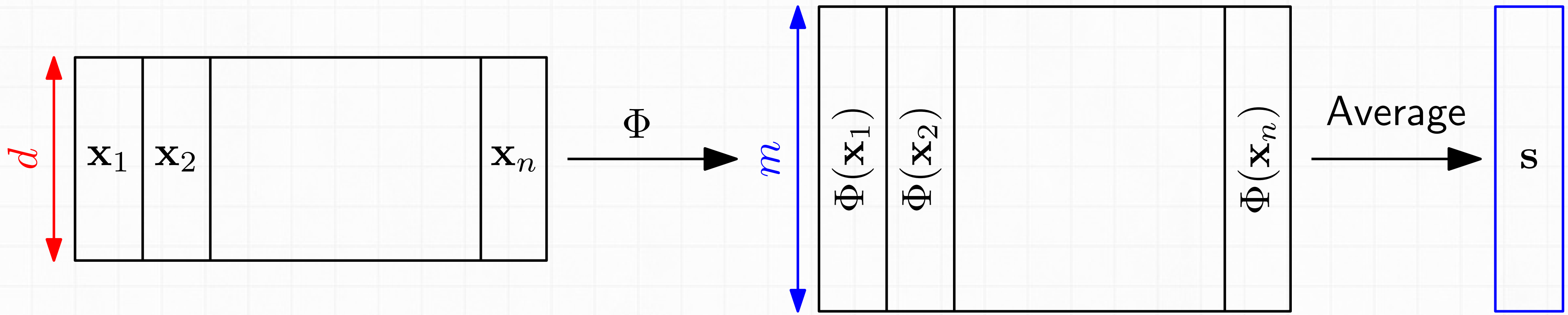
Questions :

- How to define s? (Ideally : $\|\mathbf{\Theta}\|_0 \lesssim m \ll d^2$)
- Sketch : is it efficient (computation and memory)?
- Practical decoder?
- Is it efficient?



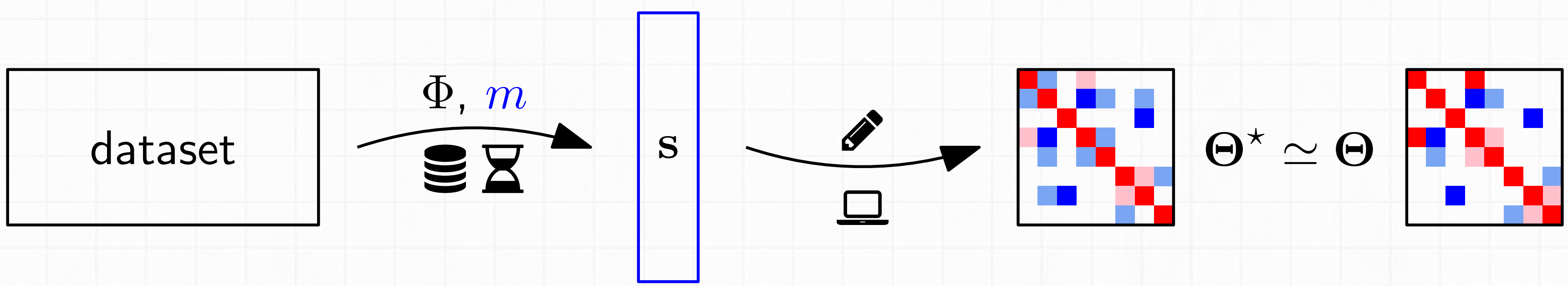
[GCK+21] Gribonval *et al.*, *Sketching Data Sets for Large-Scale Learning: Keeping only what you need*, (2021)

Sketching approach (compression) [GCK+21]



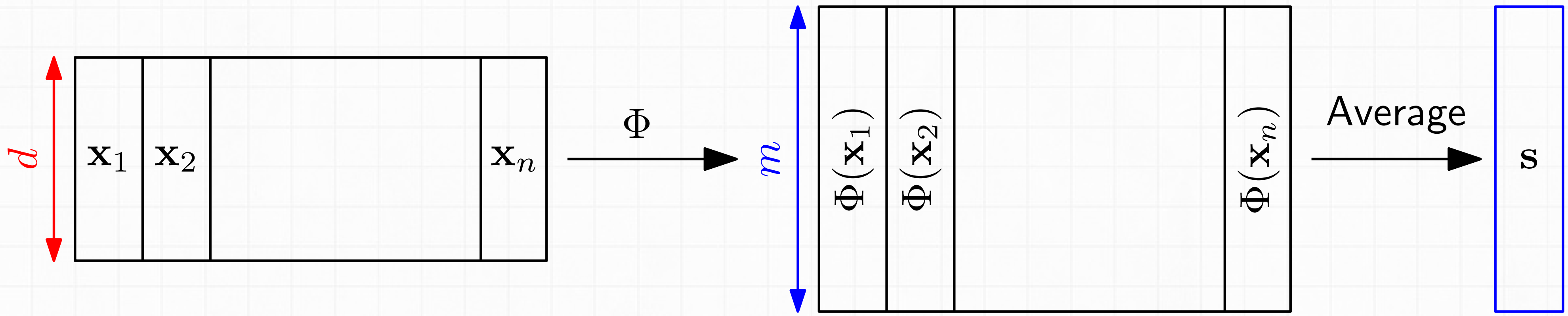
Questions :

- How to define s? (Ideally : $\|\boldsymbol{\Theta}\|_0 \lesssim m \ll d^2$)
- Sketch : is it efficient (computation and memory)?
- Practical decoder?
- Is it efficient?



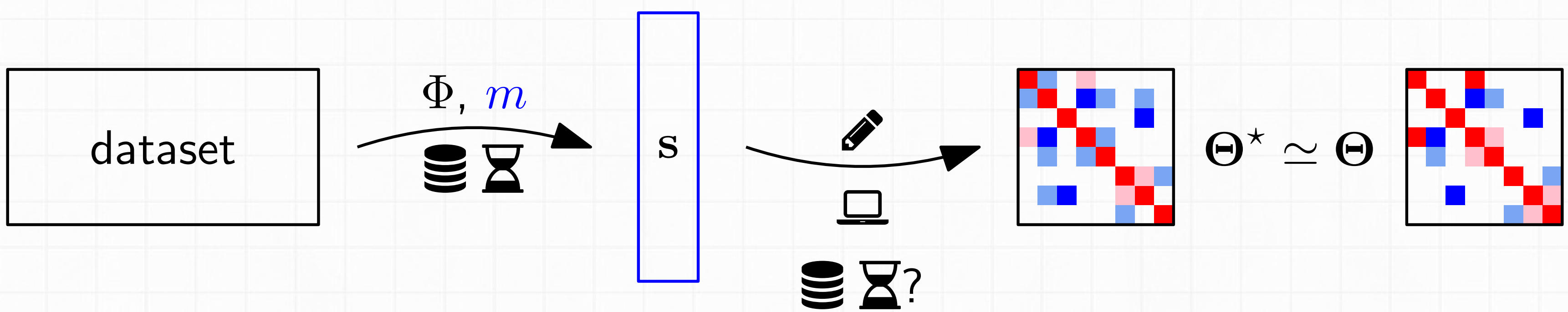
[GCK+21] Gribonval *et al.*, *Sketching Data Sets for Large-Scale Learning: Keeping only what you need*, (2021)

Sketching approach (compression) [GCK+21]



Questions :

- How to define $\mathbf{s}$? (Ideally : $\|\boldsymbol{\Theta}\|_0 \lesssim m \ll d^2$)
- Sketch : is it efficient (computation and memory)?
- Practical decoder?
- Is it efficient?



[GCK+21] Gribonval *et al.*, *Sketching Data Sets for Large-Scale Learning: Keeping only what you need*, (2021)

Random Rank-One Projections (ROP)

$$\Phi(\mathbf{x}) \triangleq m^{-1}\left(\langle \mathbf{A}_1, \mathbf{x}\mathbf{x}^\top\rangle, \ldots, \langle \mathbf{A}_m, \mathbf{x}\mathbf{x}^\top\rangle\right)^\top$$

with $\mathbf{A}_1, \ldots, \mathbf{A}_m$ random $d \times d$ matrices.

$$\mathbf{s} = \tfrac{1}{n}\sum_{i=1}^{n} \Phi(\mathbf{x}_i) = m^{-1}(\langle \mathbf{A}_j, \widehat{\boldsymbol{\Sigma}}\rangle)_j = \mathcal{A}(\widehat{\boldsymbol{\Sigma}})$$

Noisy linear measurement of $\boldsymbol{\Sigma}$.
(Compressed Sensing, Inv. Pb)

Random Rank-One Projections (ROP)

$$\Phi(\mathbf{x}) \triangleq m^{-1} \left(\langle \mathbf{A}_1, \mathbf{x}\mathbf{x}^\top \rangle, \ldots, \langle \mathbf{A}_m, \mathbf{x}\mathbf{x}^\top \rangle \right)^\top$$

with $\mathbf{A}_1, \ldots, \mathbf{A}_m$ random $d \times d$ matrices.

$$\mathbf{s} = \frac{1}{n} \sum_{i=1}^{n} \Phi(\mathbf{x}_i) = m^{-1}(\langle \mathbf{A}_j, \widehat{\boldsymbol{\Sigma}} \rangle)_j = \mathcal{A}(\widehat{\boldsymbol{\Sigma}})$$

Noisy linear measurement of $\boldsymbol{\Sigma}$.
(Compressed Sensing, Inv. Pb)

Computational and memory cost of $\Phi(x)$:

	Comp.	Memory
$\mathbf{A}_j$ full rank	$\mathcal{O}(md^2)$	$\mathcal{O}(md^2)$
$\mathbf{A}_j$ rank one $\mathbf{A}_j = \mathbf{a}_j \mathbf{a}_j^\top$	$\mathcal{O}(md)$	$\mathcal{O}(md)$

[CZ15]
[CCG15]

$$\Phi(\mathbf{x}) = m^{-1} \left(|\mathbf{a}_1^\top \mathbf{x}|^2, \; \cdots, |\mathbf{a}_m^\top \mathbf{x}|^2 \right)^\top$$

[CZ15] Cai and Zhang, *Rop: Matrix recovery via rank-one prokections*, (2015)
[CCG15] Chen *et al.*, *Exact and stable covariance estimation from quadratic sampling via convex programming*, (2015)

Borrowing from Compressed Sensing [FR13]

$$\mathbf{s} = \frac{1}{n}\sum_{i=1}^{n} \Phi(\mathbf{x}_i) = m^{-1}(\langle \mathbf{A}_j, \widehat{\boldsymbol{\Sigma}}\rangle)_j = \mathcal{A}(\widehat{\boldsymbol{\Sigma}})$$

Definition: Restricted Isometry Property (RIP).

The *sketching operator* $\mathcal{A}$ satisfies $\mathrm{RIP}(\delta, \mathfrak{S})$, if for every $\underline{\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2 \in \mathfrak{S}} \subseteq S_d$,

$$(1-\delta)\|\boldsymbol{\Sigma}_1 - \boldsymbol{\Sigma}_2\| \leq \|\mathcal{A}(\boldsymbol{\Sigma}_1) - \mathcal{A}(\boldsymbol{\Sigma}_2)\| \leq (1+\delta)\|\boldsymbol{\Sigma}_1 - \boldsymbol{\Sigma}_2\|\,.$$

[FR13] Foucart and Rauhut, *A Mathematical Introduction to Compressive Sensing*, (2013)

Borrowing from Compressed Sensing [FR13]

$$\mathbf{s} = \frac{1}{n}\sum_{i=1}^{n} \Phi(\mathbf{x}_i) = m^{-1}(\langle \mathbf{A}_j, \widehat{\boldsymbol{\Sigma}}\rangle)_j = \mathcal{A}(\widehat{\boldsymbol{\Sigma}})$$

Definition: Restricted Isometry Property (RIP).

The *sketching operator* $\mathcal{A}$ satisfies $\mathrm{RIP}(\delta, \mathfrak{S})$, if for every $\underline{\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2 \in \mathfrak{S}} \subseteq S_d$,

$$(1-\delta)\|\boldsymbol{\Sigma}_1 - \boldsymbol{\Sigma}_2\| \leq \|\mathcal{A}(\boldsymbol{\Sigma}_1) - \mathcal{A}(\boldsymbol{\Sigma}_2)\| \leq (1+\delta)\|\boldsymbol{\Sigma}_1 - \boldsymbol{\Sigma}_2\| \,.$$

Consequence :

"Optimal Decoder" : $\boldsymbol{\Sigma}^\star \triangleq \underset{\tilde{\boldsymbol{\Sigma}} \in \mathfrak{S}}{\arg\min} \|\mathcal{A}(\tilde{\boldsymbol{\Sigma}}) - \mathbf{s}\|$

$\mathfrak{S}_{k,a,b} \triangleq \{\boldsymbol{\Sigma} \succ 0,\ \boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1}$ is $(d+2k)$-sparse, $\mathrm{spec}(\boldsymbol{\Theta}) \subseteq [a,b]\}$.
$\|\mathcal{A}(\mathbf{M})\| = \|\mathcal{A}(\mathbf{M})\|_1$ and $\|\mathbf{M}\|$ defined from $\mathcal{L}(\mathbf{a})$ s.t. $\mathbb{E}\left[\|\mathcal{A}(\mathbf{M})\|_1\right] = \|\mathbf{M}\|$.

$$\|(\boldsymbol{\Sigma}^\star)^{-1} - \boldsymbol{\Theta}\|_{\mathrm{Fro}} \leq Cb^2 d\|\boldsymbol{\Sigma}^\star - \boldsymbol{\Sigma}\| \leq \frac{2Cb^2 d}{(1-\delta)}\|\mathcal{A}(\widehat{\boldsymbol{\Sigma}}) - \mathcal{A}(\boldsymbol{\Sigma})\|_1 \,.$$

[FR13] Foucart and Rauhut, *A Mathematical Introduction to Compressive Sensing*, (2013)

Theoretical recovery guarantees

$\mathfrak{S}_{k,a,b} \triangleq \{\mathbf{\Sigma} \succ 0,\ \mathbf{\Theta} = \mathbf{\Sigma}^{-1}$ is $(d+2k)$-sparse, $\mathrm{spec}(\mathbf{\Theta}) \subseteq [a,b]\}$.
$\|\mathcal{A}(\mathbf{M})\| = \|\mathcal{A}(\mathbf{M})\|_1$ and $\|\mathbf{M}\|$ defined from $\mathcal{L}(\mathbf{a})$ s.t. $\mathbb{E}\left[\|\mathcal{A}(\mathbf{M})\|_1\right] = \|\mathbf{M}\|$.

Theorem : [VLGG23]

Let $\mathcal{A}$ be our sketching operator and $\mathbf{a}_1, \ldots, \mathbf{a}_m \overset{i.i.d.}{\sim}$ Gaussian or Unif. on S_d.
$\forall \delta \in]0,1[$, $\exists C = C(\delta, b/a)$ s.t. whenever

$$m \geq C(d+2k)\log d\,,$$

$\mathcal{A}$ satisfies $\mathrm{RIP}(\delta, \mathfrak{S}_{k,a,b})$ with high probability.

[VLGG23] Vayer, **L.**, Gribonval, Gonçalves, *Compressive Recovery of Sparse Precision Matrices*, (2023)
(arXiv:2311.04673)

Theoretical recovery guarantees

$\mathfrak{S}_{k,a,b} \triangleq \{\boldsymbol{\Sigma} \succ 0,\ \boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1}$ is $(d+2k)$-sparse, $\mathrm{spec}(\boldsymbol{\Theta}) \subseteq [a,b]\}$.
$\|\mathcal{A}(\mathbf{M})\| = \|\mathcal{A}(\mathbf{M})\|_1$ and $\|\mathbf{M}\|$ defined from $\mathcal{L}(\mathbf{a})$ s.t. $\mathbb{E}\left[\|\mathcal{A}(\mathbf{M})\|_1\right] = \|\mathbf{M}\|$.

Theorem : [VLGG23]

Let $\mathcal{A}$ be our sketching operator and $\mathbf{a}_1, \dots, \mathbf{a}_m \overset{i.i.d.}{\sim}$ Gaussian or Unif. on S_d.
$\forall \delta \in]0,1[$, $\exists C = C(\delta, b/a)$ s.t. whenever

$$m \geq C(d+2k)\log d\,,$$

$\mathcal{A}$ satisfies $\mathrm{RIP}(\delta, \mathfrak{S}_{k,a,b})$ with high probability.

$$\mathfrak{S}_{k,a,b} \longleftrightarrow \mathfrak{S}_{k,\kappa_0} \triangleq \{\boldsymbol{\Sigma} \succ 0,\ \boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1} \text{ is } (d+2k)\text{-sparse},\ \kappa(\boldsymbol{\Theta}) \leq \kappa_0\}\,,$$
$$C(\delta, b/a) \longleftrightarrow C(\delta, \kappa_0)$$

[VLGG23] Vayer, **L.**, Gribonval, Gonçalves, *Compressive Recovery of Sparse Precision Matrices*, (2023) (arXiv:2311.04673)

Practical limitations

Encoding

$$\boxed{m \geq C(d + 2k) \log d}$$

	Comp.	Memory
$\mathbf{A}_j$ full rank	$\mathcal{O}(md^2)$	$\mathcal{O}(md^2)$
$\mathbf{A}_j$ rank one $\mathbf{A}_j = \mathbf{a}_j\mathbf{a}_j^\top$	$\mathcal{O}(md) = \mathcal{O}(d^2 \log d)$	$\mathcal{O}(md) = \mathcal{O}(d^2 \log d)$

Decoding

"Optimal Decoder":

$$\mathbf{\Sigma}^\star \triangleq \operatorname*{arg\,min}_{\tilde{\mathbf{\Sigma}} \in \mathfrak{S}} \|\mathcal{A}(\tilde{\mathbf{\Sigma}}) - \mathbf{s}\|$$

(Unsolvable)

Practical limitations

Encoding

$$m \geq C(d + 2k) \log d$$

	Comp.	Memory
$\mathbf{A}_j$ full rank	$\mathcal{O}(md^2)$	$\mathcal{O}(md^2)$
$\mathbf{A}_j$ rank one $\mathbf{A}_j = \mathbf{a}_j\mathbf{a}_j^\top$	$\mathcal{O}(md) = \mathcal{O}(d^2 \log d)$	$\mathcal{O}(md) = \mathcal{O}(d^2 \log d)$
Structured $\mathbf{A}_j$	$\mathcal{O}(m \log d)$	$\mathcal{O}(m)$

Decoding

"Optimal Decoder":

$$\mathbf{\Sigma}^\star \triangleq \underset{\tilde{\mathbf{\Sigma}} \in \mathfrak{S}}{\arg\min} \, \|\mathcal{A}(\tilde{\mathbf{\Sigma}}) - \mathbf{s}\|$$

(Unsolvable)

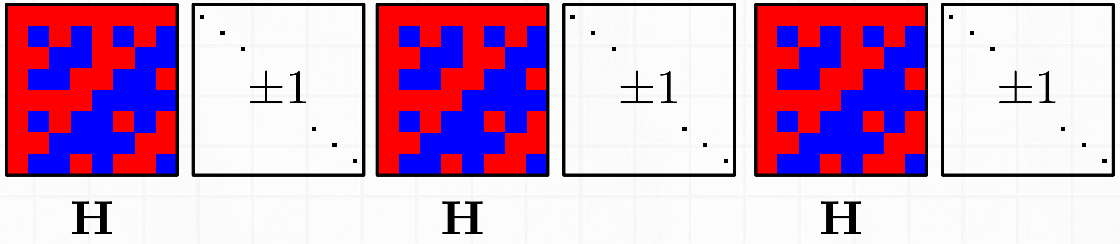
$\longrightarrow$ Iterative algorithm
(inspired by *Proximal Gradient Descent*)

In practice (Encoding)

Reformulate : $\Phi(\mathbf{x}) = \frac{1}{m}(\mathbf{A}\mathbf{x})^{\odot 2}$

$$\mathbf{A} = \begin{pmatrix} \vdots \\ \mathbf{a}_j^\top \\ \vdots \end{pmatrix} = \begin{pmatrix} \mathbf{B}_1 \\ \vdots \\ \mathbf{B}_b \end{pmatrix}$$

$(\mathbf{a}_j)\ i.i.d$ $\qquad$ $(\mathbf{B}_i)\ i.i.d$

$\mathbf{B}_i = d^{-\frac{3}{2}}$ 

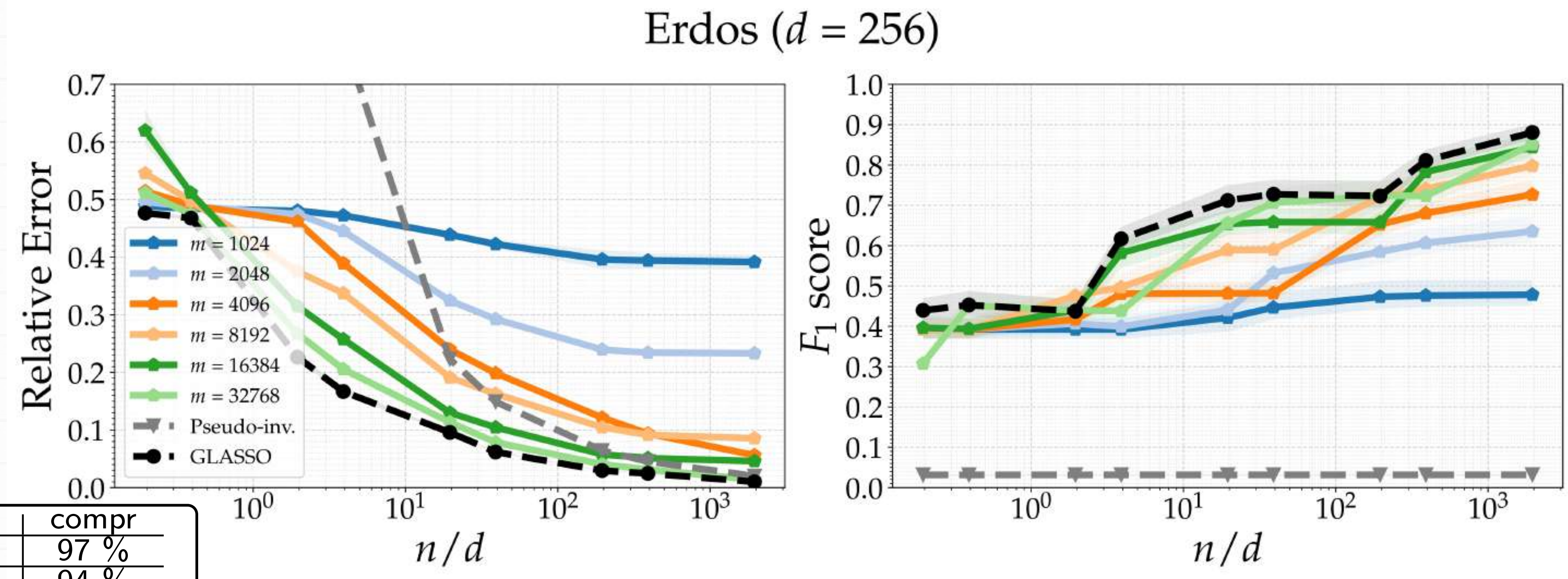
Hadamard matrix : $\mathbf{H} \triangleq \mathbf{H}_d$,

$$\mathbf{H}_1 = (1), \quad \mathbf{H}_{2k} = \begin{pmatrix} \mathbf{H}_k & \mathbf{H}_k \\ \mathbf{H}_k & -\mathbf{H}_k \end{pmatrix}$$

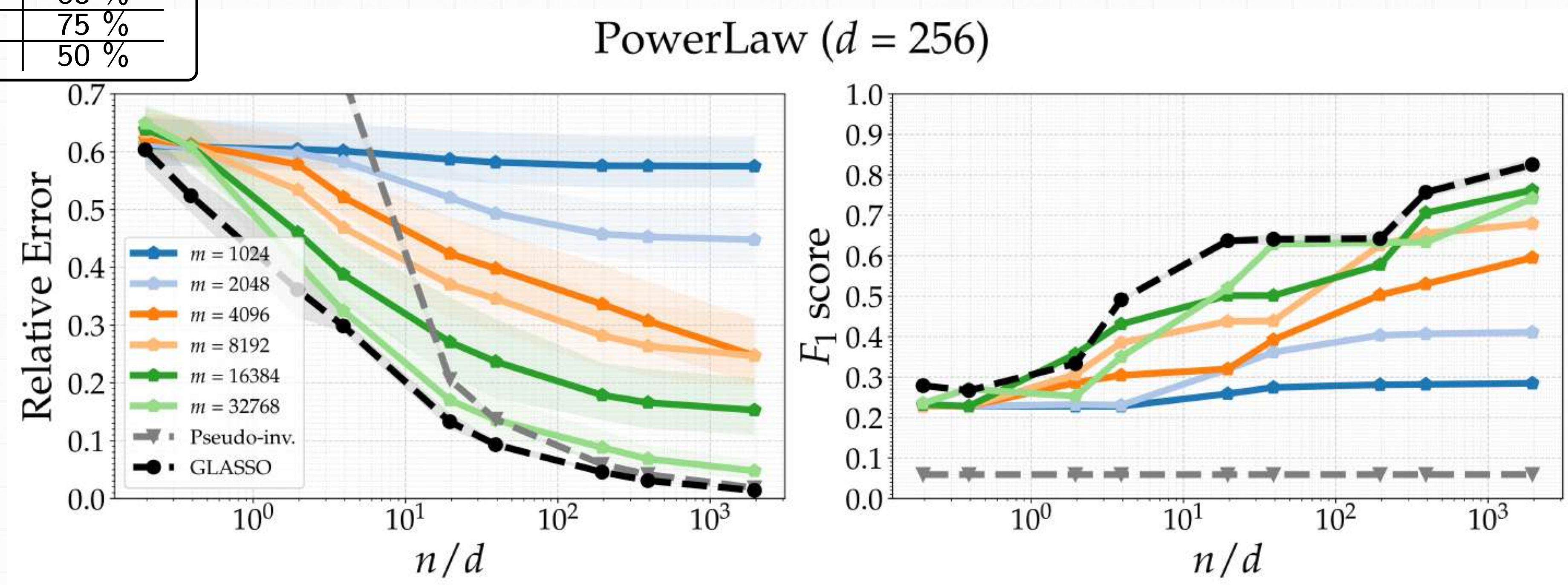
$\mathbf{H}x$ can be computed in $\mathcal{O}(d \log d)$, without storing $\mathbf{H}$ (fast Hadamard Transform).

	Comp.	Memory
Structured $\mathbf{A}$	$\mathcal{O}(m \log d)$	$\mathcal{O}(m)$

Experiments



m	compr
1024	97 %
2048	94 %
4096	88 %
8192	75 %
16384	50 %



Conclusion

Perspectives:

- Extending the theory to *structured* sketching operators.
- Guarantees for the practical decoder? (convergence? to what?)
- A more efficient practical decoder?

- Can we recover only properties of the graphs (*e.g.*, clusters)

Preprint:

Compressive Recovery of Sparse Precision Matrices,
Vayer, **L.**, Gribonval, Gonçalves (2023)
(arXiv:2311.04673)

Conclusion

Perspectives:

- Extending the theory to *structured* sketching operators.
- Guarantees for the practical decoder? (convergence? to what?)
- A more efficient practical decoder?

- Can we recover only properties of the graphs (*e.g.*, clusters)

Preprint:

Compressive Recovery of Sparse Precision Matrices,
Vayer, **L.**, Gribonval, Gonçalves (2023)
(arXiv:2311.04673)

Thank you for your attention.

In practice (Decoding)

A more practical decoding scheme:

$f(\boldsymbol{\Sigma}) \triangleq \frac{1}{2}\|\mathcal{A}(\boldsymbol{\Sigma}) - \mathbf{s}\|_2^2$ (sketch fidelity)

$$\boldsymbol{\Sigma}_0 \succ 0$$

$$\boldsymbol{\Sigma}_{t+\frac{1}{2}} = \boldsymbol{\Sigma}_t - \gamma \nabla f(\boldsymbol{\Sigma}_t) \qquad \text{(gradient descent step)}$$

$$\boldsymbol{\Sigma}_{t+1} = \mathrm{GLASSO}_{\gamma\lambda}[\boldsymbol{\Sigma}_{t+\frac{1}{2}}] \qquad \text{(denoising)}$$

In practice (Decoding)

A more practical decoding scheme:

$$f(\mathbf{\Sigma}) \triangleq \frac{1}{2}\|\mathcal{A}(\mathbf{\Sigma}) - \mathrm{s}\|_2^2 \text{ (sketch fidelity)}$$

$$\boxed{\begin{aligned}&\mathbf{\Sigma}_0 \succ 0 \\ &\mathbf{\Sigma}_{t+\frac{1}{2}} = \mathbf{\Sigma}_t - \gamma\nabla f(\mathbf{\Sigma}_t) && \text{(gradient descent step)} \\ &\mathbf{\Sigma}_{t+1} = \mathrm{GLASSO}_{\gamma\lambda}[\mathbf{\Sigma}_{t+\frac{1}{2}}] && \text{(denoising)}\end{aligned}}$$

Link with (Bregman) Proximal Gradient descent:

$$\arg\min_x \ \{f(x) + g(x)\}$$

$$\begin{aligned}&\text{init. } x_0 \\ &x_{t+\frac{1}{2}} = x_t - \gamma\nabla f(x_t) && \text{(gradient descent step)} \\ &x_{t+1} = \mathrm{prox}_g(x_{t+\frac{1}{2}}) && \text{(proximal step)}\end{aligned}$$

$$\triangleq \arg\min_u \ \left\{g(u) + \frac{1}{2}\|u - x_{t+\frac{1}{2}}\|_2^2\right\}$$

In practice (Decoding)

A more practical decoding scheme:

$$f(\boldsymbol{\Sigma}) \triangleq \frac{1}{2}\|\mathcal{A}(\boldsymbol{\Sigma}) - \mathrm{s}\|_2^2 \quad \text{(sketch fidelity)}$$

$$\boxed{\begin{array}{ll} \boldsymbol{\Sigma}_0 \succ 0 & \\ \boldsymbol{\Sigma}_{t+\frac{1}{2}} = \boldsymbol{\Sigma}_t - \gamma \nabla f(\boldsymbol{\Sigma}_t) & \text{(gradient descent step)} \\ \boldsymbol{\Sigma}_{t+1} = \mathrm{GLASSO}_{\gamma\lambda}[\boldsymbol{\Sigma}_{t+\frac{1}{2}}] & \text{(denoising)} \end{array}}$$

Link with (Bregman) Proximal Gradient descent:

$$\arg\min_x \ \{f(x) + g(x)\}$$

$$\begin{array}{ll} \text{init. } x_0 & \\ x_{t+\frac{1}{2}} = x_t - \gamma \nabla f(x_t) & \text{(gradient descent step)} \\ x_{t+1} = \mathrm{prox}_g^h(x_{t+\frac{1}{2}}) & \text{(Bregman proximal step)} \end{array}$$

$$\triangleq \arg\min_u \ \left\{ g(u) + D_h(u|x_{t+\frac{1}{2}}) \right\}$$

$$\mathrm{GLASSO}_\lambda[\mathbf{Z}] \triangleq \arg\min_{\boldsymbol{\Sigma}\succ 0} \ \{\lambda\|\boldsymbol{\Sigma}^{-1}\|_{1,\text{off}} + D_h(\mathbf{Z}|\boldsymbol{\Sigma}) \qquad \text{with } h(\mathbf{X}) = -\log\det\mathbf{X}\}$$

Recipe of the Proof

Rewrite the RIP:

$$\mathrm{RIP}(\delta, \mathfrak{S}_{k,a,b}) \Leftrightarrow \left| \|\mathcal{A}(\mathbf{U})\|_1 - 1 \right| \leq \delta\,, \quad \forall \mathbf{U} \in S[\mathfrak{S}_{k,a,b}]$$

$$\text{Normalized secant}: \; S[\mathfrak{S}_{k,a,b}] \triangleq \left\{ \tfrac{\mathbf{\Sigma}_1 - \mathbf{\Sigma}_2}{\|\mathbf{\Sigma}_1 - \mathbf{\Sigma}_2\|},\; \mathbf{\Sigma}_1, \mathbf{\Sigma}_2 \in \mathfrak{S}_{k,a,b} \right\}$$

Goal: $$\mathbb{P}\left(\left| \|\mathcal{A}(\mathbf{U})\|_1 - 1 \right| \leq \delta\,, \quad \forall \mathbf{U} \in S[\mathfrak{S}_{k,a,b}] \right) \geq 1 - \rho$$

Ingredients:

- Pointwise concentration :

$$\forall \mathbf{U} \in S[\mathfrak{S}_{k,a,b}],\ \forall t > 0,\ \mathbb{P}\left(\left| \|\mathcal{A}(\mathbf{U})\|_1 - 1 \right| > t \right) \leq C(t)$$

- Control of covering numbers:

$$\mathcal{N}(S[\mathfrak{S}_{k,a,b}], \|\cdot\|_\Lambda, \varepsilon) \triangleq \#\text{balls of size } \varepsilon \text{ to cover } S[\mathfrak{S}_{k,a,b}]$$

Experiments

$d = 64$, $m = 1536$ (compression 25%)